Listing of Claims

Please amend the claims as follows:

1. [Cancelled]

2. [Currently Amended] The method of claim 34, wherein each of the ~~one or more~~ data sequences has a length of one byte.

3. [Currently Amended] The method of claim 34, wherein each of the ~~one or more~~ data sequences has a length of one bit.

4. [Currently Amended] The method of claim 34, wherein the ~~the~~ instruction received from the processor is a member of an extensible instruction set.

5. [Currently Amended] The method of claim 34, wherein each of the one or more unaligned data sequences has a length less than a length of the second aligned word.

6. [Previously Presented] The method of claim 34, further including changing a memory address pointer by an amount less than a length of the second aligned word to point to a next unaligned data sequence to be read.

7 – 16. [Cancelled]

17. [Currently Amended] A system comprising:

~~a load/store buffer configured to store data; and~~

a processor configured to execute one or more data sequences;

an extension adapter including a load/store buffer, the extension adapter configured to execute GET instructions for processing data sequences, the GET instructions comprising the steps:

initializing the load/store buffer by loading a first aligned word into the load/store buffer from a memory;

further initializing the load/store buffer by loading a second aligned word from the memory into the load/store buffer;

reading one or more data sequences from the load/store buffer into a register file configured for instruction execution using an instruction received from the processor, at least one of the one or more data sequences being unaligned relative to the memory; and

loading additional aligned words to the load/store buffer from the memory to replace the one or more data sequences that are read from the load/store buffer into the register file.; and

a programmable instruction set extension fabric for enabling instruction extensions to be stored and executed.

18. [Currently Amended] The system of claim 17 in which the one or more number of data sequences read is an immediate specified number.

19. [Currently Amended] The system of claim 17 in which the one or more number of data sequences read is a specified number stored as an index in a register memory.

20. [Previously Presented] The system of claim 17 in which a first of the one or more data sequences read is located at a first memory location and the one or more data sequences comprises a specified number of data sequences stored as a first index in a register memory, wherein a subsequent data sequence following a first of the data sequences is located at a second memory location pointed to by a second index.

21 - 23. [Cancelled]

24. [Currently Amended] A method of processing data, the method comprising:

receiving a sequence of aligned data at an extension adapter, the receipt of the sequence controlled by a processor using a load/store instruction;

receiving a user defined instruction at the extension adapter from the processor;

loading ~~an~~ unaligned data which is a subset of the sequence of aligned data into a programmable instruction set extension fabric; and

executing the user defined instruction using the unaligned data ~~subset as an operand.~~

25. [Currently Amended] The method of claim 24, further including stalling execution of the user defined instruction to assure availability of the unaligned datasubset.

26. [Currently Amended] The method of claim 24, further including controlling the timing of the loading of the unaligned datas ubset as a function of instruction execution time.

27. [Currently Amended] The method of claim 24, further including operating on both aligned and unaligned dataoperands using the programmable instruction set extension fabric, the alignment being relative to a word size of the processor.

28. [Currently Amended] The method of claim 24, wherein loading the unaligned datasubset of the sequence into the programmable instruction set extension fabric is accomplished using an instruction configured to read data of a first size and to increment a memory address pointer by an amount different from the first size.

29. [Currently Amended] The method of claim 24, further including initializing a data pointer in order to achieve an offset between a data alignment of the processor and the unaligned datasubset.

30. [Currently Amended] The method of claim 24, wherein loading unaligned data which is a subset of the sequence of aligned data loading the unaligned subset of the sequence is performed using a PUT instruction received from the processor.

31. [Currently Amended] A method of processing data, the method comprising:

receiving a sequence of aligned data, the aligned data being aligned relative to a memory;

receiving a user defined instruction;

loading ~~an~~ unaligned data which is a subset of the sequence of aligned data into a programmable instruction set extension fabric;

executing the user defined instruction using the unaligned data ~~subset as an operand~~ to create an unaligned instruction output;

receiving the unaligned instruction output;

aligning unaligned instruction output to the memory to create aligned data; and

storing the aligned data in the memory.

32. [Currently Amended] The method of claim 31, wherein receiving the sequence of aligned data is controlled by a processor using a load/store instruction, and the user defined instruction is received from the ~~same~~ processor.

33. [Currently Amended] The method of claim 31, wherein the sequence of aligned data is received from a first location within the memory and the aligned data is written to a second different location within the memory.

34. [Currently Amended] A method for processing data sequences in a computing system, the method comprising:

initializing a load/store buffer in an extension adapter by loading a first aligned word into a load/store buffer;

further initializing the load/store buffer by loading a second aligned word into the load/store buffer, alignment of the first aligned word and the second aligned word being relative to a memory accessible via a processor;

reading an~~one or more~~ unaligned data sequence from the load/store buffer into a register file of an extension adapter for use by an instruction received from the processor, the unaligned data sequence including at least part of the second aligned word; ~~and~~

loading additional aligned words to the load/store buffer to replace the first aligned word and the second aligned word; and

executing the unaligned data sequence.

35 - 36. [Canceled]

37. [Previously Presented] The system of claim 17, wherein a length of at least one of the one or more data sequences read from the load/store buffer into the register file is less than a length of the first aligned word.

38. [Canceled]